Docket No.: 826.1721

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re the Application of:

Hironori YAHAGI

Serial No. 09/819,729 Group Art Unit: 2152

Confirmation No. 4351

Filed: March 29, 2001 Examiner: Kristina Honeycutt

For: APPARATUS CONVERTING A STRUCTURED DOCUMENT HAVING A HIERARCHY

APPEAL BRIEF

Mail Stop Appeal Brief-Patents
Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the Final Office Action in the above-identified application, and pursuant to the Notice of Appeal filed October 13, 2006, Applicants submit this Brief with the fee of $500.00 set forth by 1.17(c).

(I) Real Party In Interest

The real party in interest in this appeal is the assignee Fujitsu Limited.

(II) Related Appeals and Interferences

The undersigned attorney, the appellant and the assignee know of no related appeals or interferences which would be directly affected by or directly affect or have a bearing on the Board's decision in this appeal.

(III) Status of Claims

Claims 1-14 are pending and stand finally rejected.

(IV) Status of Amendments

No amendments have been filed subsequent to the final rejection.

(V) Summary of the Claimed Subject Matter

The present inventions are designed to solve a problem with structured documents, such as XML (Extensible Markup Language) documents. These documents are stored in the form of a hierarchical set of records, typically in the form of a tree of records (Figure 1C). Each record includes text and the tree defines how the records are combined to form the document. For example, a document having the lyrics to a song may have a tree with a node including a first record having the text "Mary had a little lamb," and an adjacent same level node in the tree may include a second record having the text "and her fleece was white as snow". Because each record and the pointers to the records occupy memory storage space, such structured documents can occupy a large amount of working memory in a computer. The inventions of the claims are designed to compress such structured documents (or combine the contents of the records) in such a way that the text relationship of the contents of the records is preserved. In the above example, a new record having the text "Mary had a little lamb, and her fleece was white as snow" would replace the two records. This version of the structured document occupies less working memory than the prior version. In addition, because the two records have been combined into one record, the processing time for the new version of the document is less than for the prior version.

In particular, "a structured document" formed of "a set of hierarchical elements" having "records each including one text element or more" is input (see claims 1 and 11, Figures 1F, 2B - 4 & pages 4, 5, 18-21). The structured document is compressed ("compressing"- see claims 12, 13 and 14, see and compare Figures 2B and 4 with figure 7 & pages 18-24).

To compress the document a "new text element" is generated or created "by combining contents of text elements" of "two records" in a way that is designed "to preserve" or "preserving a text relationship between the contents" (see claims 1, 9, 10, 11, 12 and 13, Figures 13-17 & pages 21, 22, 27-40). A "new record" is created that "includes the new text element and inherits a relative position relationship of text elements in the two records (see claims 1, 9, 10 and 11, Figures 13-17 & pages 21, 22, 27-40).

The compressing can also be by "combining hierarchical text elements of the hierarchically structured document responsive to common element names for the hierarchical text elements and a common child element structure of the hierarchical text elements and preserving a text relationship between contents of the text elements" claim 12, Figures 5, 13-17 & pages 21, 22, 27-40) or by "combining hierarchical text elements of the hierarchically structured document responsive to common features of parts of the hierarchy of the

hierarchically structured document and preserving a text relationship between contents of the text elements" (claim 13, Figures 5, 13-17 & pages 21, 22, 27-40) or by "compressing the hierarchy of the structured document by combining text element content and preserving a hierarchically defined relationship between contents of text elements" (claim 14, Figures 5, 13-17 & pages 21, 22, 27-40).

The structured document is then converted ("converting") by "by replacing the two records or more with the new record, thereby decreasing the number of hierarchical elements of the structured document" (see claims 1, 9, 10 and 11, Figures 5, 13-17 & pages 21, 22, 27-40). And then the structured document is output " after being converted" (see claims 1 and 11, Figure 16 & page 37).

The combining can be of more than two elements "synthesis targets" and the combining can be based on a particular hierarchical structure that includes "a first combination of elements that successively exist side by side in a level immediately below a certain element and have a same element name, and content of each element included in a second combination of elements that have a same element name in a certain level lower than the elements of the first combination, elements in each level on a route from the elements of the first combination to the certain level having a same element name" (see claim 3, Figures 5, 13-17 & pages 21, 22, 27-40). This hierarchical structure results in "generating a synthesized substructure that includes the plurality of new text elements, and inherits a relative position relationship of original elements among the plurality of new text elements" (see claim 3, Figures 5, 13-17 & pages 21, 22, 27-40). The structured document can also have elements duplicated ("duplication") where the duplication is "of an unjoined element below a new element included in a synthesized substructure generated from an element higher than the unjoined element" (see claim 3, Figures 5, 13-17 & pages 21, 22, 27-40).

(VI) Grounds Of Rejection To Be Reviewed On Appeal

Claims 1-14 stand finally rejected.

Claims 1, 9-11 and 14 stand rejected under 35 USC § 103(a) over Brintzenhofe (US Pub. 20050223320) and Zeng (US Pub. 20020003906).

Claim 2 stands rejected under 35 USC § 103(a) over Brintzenhofe (US Pub. 20050223320), Zeng (US Pub 20020003906) and DeRose (US 6,105, 044).

Claims 3, 4, 12 and 13 stand rejected under 35 USC § 103(a) over Brintzenhofe (US Pub. 20050223320), Zeng (US Pub 20020003906) and Ardoin (US Pub. 20020099684).

Claim 5 stands rejected under 35 USC § 103(a) over Brintzenhofe (US Pub. 20050223320), Zeng (US Pub 20020003906) and Alam (US 6,336,124).

Claims 6-8 stand rejected under 35 USC § 103(a) over Brintzenhofe (US Pub. 20050223320), Zeng (US Pub 20020003906), Ardoin (US Pub. 20020099684) and DeRose (US 6,105, 044).

(VII) Argument

As noted above, the present inventions are designed to solve a space utilization problem with structured documents stored in the form of a hierarchical set of records that include text where the hierarchy defines how the records are combined to form the document. Because each record occupies memory storage space, such structured documents can occupy a large amount of working memory in a computer. The inventions of the claims are designed to compress such structured documents (or combine the contents of the records) in such a way that the text relationship of the text contents of the records is preserved. The compressed structured document occupies less working memory and the processing time of the compressed structured the document is reduced as compared to the uncompressed document.

Claims 1, 9-11 and 14

The Examiner rejects independent claims 1, 9-11 and 14 as obvious over Brintzenhofe and Zeng.

Brintzenhofe discusses a system for defining a document that allows the document to be rendered (or reproduced) for multiple forms of output media. That is, Brintzenhofe is about generating documents not compressing a structure of a document. Brintzenhofe separates the information about a document into content, design and media aspects and represents the content.

Zeng is about adaptive coding of images. In Zeng an image is divided into segments. Image segmentation can apparently allow more efficient image coding. An image filter is assigned to each segment because the filters (A-C) are designed for different types of image content, such as sharp edges or slowly varying image changes. Because neighboring segments of the image can be processed using the same filter, the Zeng system can re-segment the image based on the common filter and assign the common filter to the re-segmentation. Once the segments are finalized the image is parsed based on the segmentation and the image segments are coded using the assigned filters.

It is submitted that Brintzenhofe and Zeng are non-analogous arts, that there is no suggestion or motivation to combine them and Zeng teaches away from a combination with Brintzenhofe as discussed later herein.

The Examiner admits that Brintzenhofe does not teach combining text elements and generating a new record and looks to Zeng for such teaching. In particular, he Examiner alleges that the elements A in Zeng are text elements. (We note that Examiner appears to have abandoned the interpretation that elements A are text elements and in the Advisory Action appears to assert that the binary numbers discussed in Zeng are text elements). In the final Action the Examiner particularly states:

> Brintzenhofe discloses combining elements and preserving relationships (p.7, para. 91; p.25, para. 247, 248) but does not disclose a joining device generating a new text element by combining contents of text elements relatively at a same position among two records or more of the structured document or a generating device generating a new record that includes the new element and inherits a relative position relationship of elements in two records or more. Zeng discloses combining text elements at relatively the same position, creating a new text element (Fig. 6, 7; p.2, para. 19, 20; p.3, para. 34) since all A elements on the leftmost branches of the tree of Figure 6 are combined to form a new A element in Figure 7.

(See Final Action, page 3)

> Applicants indicate that Zeng is not about structured documents much less structured documents of text (p.6, para. 5). The Examiner disagrees since Zeng teaches inputting an image including text and graphic regions and dividing the image into segments to form a tree (Abstract, p.4, para 52). Zeng further teaches combining elements in the tree based on position and similarities to create a new text element (Fig. 6, 7; p.2, para. 19, 20; p.3, para. 34) since elements with A text in the leftmost branches of the tree in Figure 6 are combined to form an A text element in Figure 7. The text relationship is preserved since all A elements in the leftmost branches of the tree in Figure 6 are combined into an A element in the tree in Figure 7.

(See Final Action, page 12)

As can be seen, the Examiner asserts that figures 6 and 7 of Zeng, particularly figure 7 show text elements "with A text in the left most branches" and shows the preservation of text relationship ("The text relationship is preserved since all A elements in the leftmost branch of the tree in Figure 6 are combined into an A element in the tree in Figure 7"). That is, the Examiner is basing the final rejection on the assertion that the "A elements" of Zeng are text. This is not correct. The interpretation by the Examiner is wrong. According to the text of Zeng, the letters A, B and C represent filters not text. The Zeng text particularly states:

> [0032] Referring now to FIGS. 6 and 7, an illustration of joint segmentor/filter selector operation is presented. In FIG. 6, block 80 represents an input image that has been initially segmented into 16 equally-sized **subregions (numbered**

0000 through 1111 binary) by a quadtree segmentor. Quadtree 82 shows the relationship of the segments in the quadtree mapping of segmented image 80.
[0033] Wavelet filters A, B, and C are applied to each **subregion 0000-1111**, and a best filter is chosen for each subregion based on entropy estimation. A sample filter assignment is shown on 80 by the placement of "A", "B", or "C" on the subregion to show the assigned filter. The leaf nodes on quadtree 82 are also labeled with the selected filter for that node.
[0034] After all leaf subregions have been assigned a filter type A, B, or C, the tree is merged from the bottom up recursively. If the children of a parent node are assigned the same type of filter, the children will be merged and the same type of filter will be assigned to the parent node. FIG. 7 shows the original segmentation and filter assignment of FIG. 6 after this merging process. Of the original leaf segments of quadtree 82, **only segments 1000 and 1001** remain in quadtree 86. All other original leaf segments have been merged up one or two nodes due to common filter assignment. Segmentation map 84 shows the final filter assignment for this example.

(See Zeng, paragraphs. 32-34, underlining and **bold** emphasis supplied)

The above text, particularly that underlined for emphasis, discusses assigning filters to image segments in a hierarchy, not assigning text in a hierarchical relationship to preserve the text relationship.

In asserting an interpretation of the "elements A" as text in the final Action, the Examiner particularly points to Zeng figures 6 and 7 and paragraphs 19, 20 and 34. Paragraph 34 is set forth above. Paragraphs 19 and 20 particularly state:

[0019] FIG. 6, which illustrates a quadtree-segmented image prior to leaf node merging, and its corresponding quadtree structure; and
[0020] FIG. 7, which shows the same quadtree-segmented image after leaf node merging, and its corresponding quadtree structure.

(See Zeng, paragraphs. 19 & 20).

As can be seen from the above portions of Zeng, there is no discussion support for the "elements A" being text. Figures 6 and 7 of Zeng merely show numbers (image segment identification numbers) and letters (filter identification letters) and say nothing about text much less about the "elements A" being text.

As noted above, the Examiner appears to assert in the Advisory Action that the binary numbers in Zeng are now interpreted as text. In particular, the Examiner states:

> Regarding independent claim 1, Applicants argue that Zeng does not teach “the preservation of text relationship” (p. 7, para. 3). The Examiner disagrees because Zeng teaches generating a new text element by combining contents of text elements relatively at a same position among two records or more of the structured document to preserve a text relationship between the contents (Abstract; Figures 6, 7; p.2, para. 19, 20; p.3, para. 34; p.4, para. 52). Zeng teaches an image that includes text being divided into a tree. Using filters, the

nodes of the tree are combined to form new elements. Figure 6 represents an input image that has been segmented into subregions (p.3, para. 32). Figure 7 shows the segmentation and filter assignment of Figure 6 after the merging process (p.3, para. 34). The nodes have been merged based on shared filters (A, B, C). The text (the binary numbers 0000 through 1111) have been preserved in the tree since each 4 digit number can still reach the correct filter by traversing the tree.

Applicants further argue that Brintzenhofe and Zeng do not discuss the text compression where there is a replacement of two or more text records with a new text record (p.9, para. 2). The Examiner disagrees because the term "text compression" is not claimed in the presently presented claims. Furthermore, Zeng teaches replacing multiple records with a new record, thereby decreasing the number of hierarchical elements of the structured document (Figures 6, 7; p.3, para. 34). Figure 6 represents an input image that has been segmented into subregions (p.3, para. 32) and Figure 7 shows Figure 6 after the nodes with common filter have been merged (p.3, para. 34). The nodes have been replaced by new elements since all nodes beginning with "00" and having a filter "A" have been replaced by a single A node.

(See Advisory Action, Continuation sheet, underlining emphasis supplied)

Paragraphs 19, 20 and 32-34 of Zeng, referred to by the Examiner, are set forth above on pages 5 and 6. As can be seen from the Zeng text, particularly the text emphasized in bold, the binary numbers of Zeng are not part of the images being processed nor are they text but are rather image segment identification numbers. As a result, there is no teaching in Zeng of preserving text much less one of preserving a text relationship.

In fact a visual inspection of Zeng figures 6 and 7 indicates that the alleged text (binary numbers) is not preserved but actually destroyed. In looking at the segmented 80 image in Figure 6, we see on the left hand side, in association with filter identifier "A", the binary numbers 0000 0001 0010 0011. If these binary numbers were text and the text and the text relationship were preserved, we would expect to see this same text in the same position in the segmented image 84 of Zeng figure 7. In fact what we see is the binary number 00. The entire string 0000 0001 0010 0011 of figure 6 has been lost (destroyed). That is, not only is the alleged text destroyed but also the relationship between the text is destroyed. If the image segments containing the alleged text were to be considered text records, the text of the alleged record containing 0001 would be destroyed; and the relationship between the alleged record containing the alleged text 0010 and the record containing the alleged text 0011 would be destroyed because the alleged text itself is destroyed. We do not see anything in Zeng like a first record - "Mary had a little lamb," being combined with a second record - "and her fleece was white as snow" producing a new record - "Mary had a little lamb, and her fleece was white as snow".

The Examiner also acknowledged on page 3 of the final Action that Brintzenhofe does not teach replacing records with a new record and looks to Zeng for such teaching. The Examiner appears to be equating the nodes in Zeng with records. This interpretation that Zeng is teaching records appears to be reasserted in the Advisory Action as noted in the Advisory Action text set forth above. In particular, in the final Action, the Examiner stated:

> Brintzenhofe discloses converting a document and outputting the structured document (p.1, para. 6; p.4, para. 68) but does not disclose converting the structured document by replacing the two records or more with the new record, thereby decreasing the number of hierarchical elements of the structured document. Zeng discloses replacing multiple records with a new record that decreases the number of hierarchical elements (Fig. 6, 7; p.3, para. 34). It would have been obvious to one of ordinary skill in the art, having the teachings of Brintzenhofe and Zeng before him at the time the invention was made, to modify document conversion as taught by Brintzenhofe to include decreasing the number of elements by replacement as taught by Zeng, because replacing multiple records with a new record, as taught by Zeng (Fig. 6, 7; p.3, para. 34), would create a more concise hierarchical structure of the document.

(See Final Action, pages 3 and 4)

The tree of Zeng does not teach records but rather **leaf nodes** of a quad tree that have filter identifiers as labels (see Zeng paragraph 33 above). While the nodes of the tree have been rearranged because the image has been re-segmented, the nodes are not records.

The Examiner also asserts, as noted in the quotations above, that the combining by Zeng is the combining of content of the alleged records. If the nodes were records, and the labels considered content, there is no combining of the content. The filter "A" is still the filter "A" after the leaf nodes that are labeled filter A are combined. No content has been combined. If the labels were content, a new alleged record that was indeed a combination might include the label A-C, but it does not. To combine such labels (alleged content) would destroy the intent of Zeng, which is to apply the best filter to each image segment (see Zeng paragraphs 6 and 8).

As can be seen from the above discussion, the rational or reasons used by the Examiner for combining Brintzenhofe and Zeng and the combination produced are faulty. For this reason, the rejection of claims 1, 9-11 and 14 should be reversed.

The rejections of claims 2-8, 12 and 13, which are also founded on the faulty reasoning, should be reversed for the same reasons.

Claim 1 particularly recites "a joining device generating a new text element by combining contents of text elements relatively at a same position among two records or more of the structured document to preserve a text relationship between the contents". Brintzenhofe does

not teach or suggest such. Zeng does not address a structured document much less text, does not address combining text elements, does not generate a new text element via the combination and does not combine the text of records at a relatively same position and does not combine them in a way that preserves the text relationship. Rather, Zeng (see paragraphs 30-35) discusses image coding and assigns filters to segments of a segmentation map of an image using a quad tree where the leaf nodes are assigned filters. Zeng then merges unnamed nodes that have a common leaf node filter to facilitate coding of the segmentation map and uses the quad tree to parse the image segments to the filters for decomposition. There is no combining of text elements of a structured document into a new text element from records at a same relative position of a structured document in a way that preserves the relationship of the of the contents of the text elements.

Claim 1 also recites "a generating device generating a new record that includes the new text element and inherits a relative position relationship of text elements in the two records or more". Brintzenhofe does not teach or suggest such. Zeng also does not do this. Rather, Zeng merges nodes that have a common leaf node filter assignment. No new record is created. Those leaf node labels that are filter A labels are still filter A labels.

Claim 1 also recites "a converting device converting the structured document by replacing the two records or more with the new record, thereby decreasing the number of hierarchical elements of the structured document". Brintzenhofe does not teach or suggest such. Zeng also does not do this. Zeng does not convert a structured document but rather codes an image divided into segments. Zeng does not replace two records with one record but rather adjusts filter assignments relative to a re-segmented image. Zeng does not decrease the number of records in a hierarchical document but rather creates an image segment map that assigns the segments to different coders for adaptive coding of the segmented image.

Claims 9-11 emphasize the features discussed above with respect to claim 1.

Claim 14 has been rejected for the same rational as claim 1. Claim 14 calls for "analyzing a hierarchy of the hierarchically structured document; and compressing the hierarchy of the structured document by combining text element content and preserving a hierarchically defined relationship between contents of text elements". Brintzenhofe does not teach or suggest such. Zeng also does not do this. Zeng does not analyze the structure of a structured document but rather assigns filters to segments of a segmented image for adaptive image coding and changes the assignments of the filters to the segments. Zeng does not compress the hierarchy of a structured document but rather merges filter assignments that have a common

filter assigned to a leaf node representing an image segment. Zeng does not combine text element content but rather merges filter assignments for image processing segments that have common filter assignments at the leaf nodes corresponding to the segments. Zeng does not preserve a hierarchical relationship between contents of text elements as Zeng does not deal with text elements but rather deals with filters assigned to image segments.

Claim 2

The Examiner rejects dependent claim 2 as obvious over Brintzenhofe, Zeng and DeRose.

First, DeRose does not cure the defects of Brintzenhofe and Zeng and thus this claim patentably distinguishes over the combination including DeRose for the reasons discussed above with respect to claim 1.

Second, as discussed above claim 1 calls for taking the contents of two records and combining or converting them to a new record. Claim 2 calls for searching for a particular character string in the structured document and "restoring a record before being converted, which includes the search key, from the detected character string and the extracted character string, and outputting the restored record as a search result". While DeRose discusses searching and formatting documents on the fly as they are rendered, DeRose says nothing about restoring records.

Claims 3, 4, 12 and 13

The Examiner rejects these claims as obvious over Brintzenhofe, Zeng and Ardoin.

First, Ardoin does not cure the defects of Brintzenhofe and Zeng and thus these claims 3, 4, 12 and 13 patentably distinguish over the combination including Ardoin for the reasons discussed above with respect to claim 1.

Claim 3 emphasizes additional features over claim 1 that particularize how the text element are combined. In looking at claim 3 it calls for combining elements with the same name that exist side by side on the same level and combining elements lower in the hierarchy that also have the same name and are on a route from the elements of the first combination. That is, the double combining is based on a particular hierarchical structure. The Examiner looks to Ardoin for a different part of claim 3, and as Brintzenhofe does not combine text elements the Examiner must look to Zeng for such. Zeng is about image coding and not about processing a structured document. Further, when looking at Zeng, the intermediate nodes the trees of figures 6 and 7 do not have element names only the leaf nodes are assigned filter names and there are no

nodes lower down than the leaf nodes and thus no lower down nodes with the same name on a route from upper nodes with the same name that can be combined. The structure being combined with nodes above and below on a route and with the same name cannot exist in Zeng. Thus, Zeng cannot teach or suggest combining such

Claim 3 also calls for "a duplicating device generating a duplication of an unjoined element below a new element included in a synthesized substructure generated from an element higher than the unjoined element". While the paragraph (573) noted by the Examiner in Ardoin does discuss copying an element, Ardoin does not discuss duplication of an element that has not been joined and that is located below a new element that was generated from an element higher that the unjoined element.

Claim 3 also calls for "a deleting device deleting an unnecessary original element". The Examiner has made no allegations concerning this feature appearing in any of the prior art. A prima facie case of obviousness has not been made.

Claim 4 depends from claim 3 and is patentable over the combination of Brintzenhofe, Zeng and Ardoin for the reasons discussed above.

Claim 4 also calls for making a decision and generating a synthesized structure when the particular structure of the document called for in claim 3 does not exist. The Examiner again looks for this structure in Zeng, and as discussed above, this structure where elements on different levels and on a route to the level have the same name does not exist in Zeng as leaf nodes (the bottom most level) are the only ones assigned filter names.

The Examiner rejects claim 12 for the same rationale as claim 3. First, nowhere in the discussion of claim 3 does the Examiner assert that Brintzenhofe, Zeng or Ardoin teach or suggest "analyzing a hierarchy of the hierarchically structured document". As a result, a prima facie case of obviousness has not been made.

Claim 12 also calls for “combining hierarchical text elements” when there are "common element names for the hierarchical text elements" and "a common child element structure of the hierarchical text elements". The Examiner has not addressed the existence of this common element name and common child element structure in the prior art and so a prima facie case of obviousness has not been made. Zeng does not address text elements. Further, nodes below the filter named leaf nodes in Zeng do not exist and thus the structure being claimed does not exist to be combined in Zeng.

The Examiner rejects claim 13 for the same rationale as claim 3. Again, nowhere in the discussion of claim 3 does the Examiner assert that Brintzenhofe, Zeng or Ardoin teach or suggest "analyzing a hierarchy of the hierarchically structured document". As a result, a prima facie case of obviousness has not been made. Zeng also does not address text elements.

Claim 13 also calls for “combining hierarchical text elements” when there are "common features of parts of the hierarchy of the hierarchically structured document". The Examiner has not addressed the existence of this feature of claim 13 in the prior art and so a prima facie case of obviousness has not been made.

Claim 5

The Examiner rejects dependent claim 5 as obvious over Brintzenhofe, Zeng, Ardoin and Alam.

Claim 5 depends from claim 3 and is patentable over the combination of Brintzenhofe, Zeng, Ardoin and Alam for the reasons discussed above with respect to claim 3. Alam does not cure the deficiency of Brintzenhofe, Zend and Ardoin. Claim 3 calls for dividing the "second combination of the elements into a plurality of groups each composed of a predetermined number of elements, and specifies the synthesis targets based on the predetermined number of elements included in each of the groups". The Examiner looks to col. 15, line 65-col. 16, line 3 of Alam for this feature. The text noted by the Examiner particularly states:

> If the current block is determined not to be within the display parameter of the display configuration at step 1808, then the current block is divided into portions such that each portion is within the display parameter of the display configuration and the portions are displayed at step 1812.

(See Alam, col. 15, line 65-col. 16, line 3)

As can be seen this text says nothing about specifying a synthesis of elements based on a number of elements in a group.

Claims 6-8

The Examiner rejects dependent claims 6-8 as obvious over Brintzenhofe, Zeng, Ardoin and DeRose.

Claim 6 depends from claim 3 and is patentable over the combination of Brintzenhofe, Zeng, Ardoin and DeRose for the reasons discussed above with respect to claim 3. DeRose does not cure the deficiency of Brintzenhofe, Zeng and Ardoin.

Claim 6 also calls for generating "contents of the new elements by inserting a delimiter between two joined contents". The Examiner points to text in DeRose that is about putting a

delimiter between a tag and an element name in a new element in a directory. This is about naming new directory elements and inserting the new elements in a directory not about combining contents. There is no motivation from this for operating with a system that joins text elements.

Claim 7 depends from claim 6, which depends from claim 3, and is patentable over the combination of Brintzenhofe, Zeng, Ardoin and DeRose for the reasons discussed above with respect to claims 3 and 6. DeRose does not cure the deficiency of Brintzenhofe, Zeng and Ardoin.

Claim 7 calls for inserting the delimiter "if content of an element which becomes the synthesis target is lacking". DeRose says nothing about this conditional inserting.

Claim 8 depends from claim 6, which depends from claim 3, and is patentable over the combination of Brintzenhofe, Zeng, Ardoin and DeRose for the reasons discussed above with respect to claims 3 and 6. DeRose does not cure the deficiency of Brintzenhofe, Zeng and Ardoin.

Claim 8, somewhat like claim 2, calls for "restoring a corresponding portion of the structured document before being converted from the detected character string and the extracted character string". As discussed above DeRose says nothing about this.

Other Considerations

Zeng is directed to a method of adaptive coding where a wavelet transform is performed. Adaptive coders that use such transforms throw information away. (See Zeng, paragraphs 2-6 and particularly paragraph 4 - "discards the remainder"). That is, when Zeng codes an image using adaptive coding, information is discarded. An inventor seeking to preserve text in a document, as is emphasized in the claims, would not look to an adaptive coding disclosure, such as provided by Zeng for a solution, because adaptive coding deliberately discards information.

Further, a teaching of deliberately discarding information when compressing or combining teaches away from preserving information as occurs in the claims of this application.

In addition Zeng teaches away from a combination that would preserve a text relationship between the elements when compressing a structured document that combines text elements. In Zeng the filters assigned to each image segment are used to produce transform coefficients. The coefficients are then combined in a composite wavelet that produces a composite coefficient image. (See Zeng Abstract) The resulting composite coefficient image 46 Zeng (figure 2B) produced by the compositing of the image coefficients overlays the coefficient images

40 and 42. That is, if the images 40 and 42 represented text, the alleged text of one image (42) would be written on top of (or overlaid on) the alleged text of another image (40). Writing this alleged text (42) on top of other alleged text (40) would not preserve a text relationship but would destroy it. Again, deliberately destroying information teaches away from preserving the information and away from preserving a relationship among such information.

It is submitted that Zeng is not in the field of endeavor of the invention claimed in the above-identified application. The field of endeavor of the claims of the present application is the field of memory space saving for a structured document. In contrast, Zeng is in the field of adaptive or joint coding of images using multiple wavelet transforms (see Zeng, paragraph 1). The problem addressed by the claims of the present application is how to save space while also preserving the text of the structured document and preserving a relationship among the text. In contrast, Zeng is directed to the non-pertinent problem of processing an image using different transform coders for different wavelet transforms to process sub regions or segments of the image (see Zeng, paragraphs 6-9). Two very different problems. It is submitted that one of ordinary skill seeking to solve the problem solved by the claims of the present invention would not look to Zeng. Zeng is from a non-analogous art.

The Examiner has used hindsight in combining Brintzenhofe and Zeng. The Examiner looked at Brintzenhofe and understood that while it is about documents, is not about compressing a structured document to decrease the number of elements of the structured document while preserving a text relationship between element contents. As Zeng is about image coding, not structured documents, the Examiner tried to force fit Zeng to what was missing from Brintzenhofe. First, the Examiner tried to do this by equating filter assignments to leaf nodes for a segmentation map to text records that contain the text content of the structured document (see final Action) to address the relationship preservation feature. When that was shown to be inappropriate the Examiner tried again (see Advisory Action) by equating node identifiers in a quad tree with text records and trying to say that node identifier string 00 somehow preserves the relationship of a node string 0000 0001 0010 0011. This attempt is again inappropriate as the relationship is not preserved but destroyed. The Board is urged to recognize this effort by the Examiner for what it is, an inappropriate hindsight reconstruction or inappropriate interpretation.

It is submitted, based on the discussions above, that the prior art does not teach or suggest the features of the claims of the above-identified application.

Conclusion

It is submitted that the Examiner has not made a prima facie case of obviousness by preponderance of the evidence and reversal of the rejection is requested.

Respectfully submitted,

STAAS & HALSEY LLP

Date: _____

By: /J. Randall Beckers/
J. Randall Beckers
Registration No. 30,358

1201 New York Avenue, NW, Suite 700
Washington, D.C. 20005
Telephone: (202) 434-1500
Facsimile: (202) 434-1501

VIII. Claims Appendix

1. A converting apparatus, comprising:

a document inputting device inputting information of a structured document that is written with a set of hierarchical elements, and composed of a plurality of records each including one text element or more;

a joining device generating a new text element by combining contents of text elements relatively at a same position among two records or more of the structured document to preserve a text relationship between the contents;

a generating device generating a new record that includes the new text element and inherits a relative position relationship of text elements in the two records or more;

a converting device converting the structured document by replacing the two records or more with the new record, thereby decreasing the number of hierarchical elements of the structured document; and

a document outputting device outputting the structured document after being converted.

2. The converting apparatus according to claim 1, further comprising:

a key inputting device inputting a search key; and

a searching device searching the structured document after being converted with the search key, extracting a character string corresponding to a position of a detected character string from contents of an element in a certain record when the character string corresponding to the search key is detected from contents of another element in the certain record, restoring a record before being converted, which includes the search key, from the detected character string and the extracted character string, and outputting the restored record as a search result.

3. A converting apparatus, comprising:

a document inputting device inputting information of a structured document written with a set of hierarchical elements;

a storing device storing the information of the structured document;

a joining device generating a plurality of new text elements by combining, as synthesis targets, content of each element included in a first combination of elements that successively exist side by side in a level immediately below a certain element and have a same element name, and content of each element included in a second combination of elements that have a same element name in a certain level lower than the elements of the first combination, elements

in each level on a route from the elements of the first combination to the certain level having a same element name, in the structured document to preserve a text relationship between the contents;

a generating device generating a synthesized substructure that includes the plurality of new text elements, and inherits a relative position relationship of original elements among the plurality of new text elements;

a duplicating device generating a duplication of an unjoined element below a new element included in a synthesized substructure generated from an element higher than the unjoined element;

a deleting device deleting an unnecessary original element;

a converting device converting the structured document into a structured document of a synthetic type configured by a synthesized substructure by using said joining device, said generating device, said duplicating device, and said deleting device, thereby decreasing the number of hierarchical elements of the structured document; and

a document outputting device outputting the structured document of the synthetic type.

4. The converting apparatus according to claim 3, wherein

said generating device generates the synthesized substructure if a combination of elements that successively exist side by side and have a same element name in two levels or more on the route to the certain level is not found.

5. The converting apparatus according to claim 3, wherein

said joining device divides the second combination of the elements into a plurality of groups each composed of a predetermined number of elements, and specifies the synthesis targets based on the predetermined number of elements included in each of the groups.

6. The converting apparatus according to claim 3, wherein

said joining device generates contents of the new elements by inserting a delimiter between two joined contents.

7. The converting apparatus according to claim 6, wherein

said joining device consecutively inserts the delimiter in the contents of the new elements if content of an element which becomes the synthesis target is lacking.

8. The converting apparatus according to claim 6, further comprising:

a key inputting device inputting a search key; and

a searching device comparing a character string between two delimiters, which is included in contents of elements within the structured document of the synthetic type, with a character string of the search key, obtaining an order of a delimiter preceding a character string corresponding to the search key when the character string corresponding to the search key is detected from contents of elements within a certain synthesized substructure, extracting a character string between a delimiter corresponding to the order and a next delimiter in contents of another element in the certain synthesized substructure, restoring a corresponding portion of the structured document before being converted from the detected character string and the extracted character string, and outputting the restored portion as a search result.

9. A computer-readable storage medium on which is recorded a program for causing a computer to execute a process, said process comprising:

generating a new text element by combining contents of text elements relatively at a same position among two records or more of a structured document to preserve a text relationship between the contents and that is written with a set of hierarchical elements and composed of a plurality of records each including one text element or more;

generating a new record that includes the new text element and inherits a relative position relationship of text elements in the two records or more, thereby decreasing the number of hierarchical elements of the structured document; and

converting the structured document by replacing the two records or more with the new record.

10. A propagation signal for propagating a program to a computer, the program causing the computer to perform:

generating a new text element by combining contents of elements relatively at a same position among two records or more of a structured document that is written with a set of hierarchical elements to preserve a text relationship between the contents and composed of a plurality of records each including one element or more;

generating a new record that includes the new text element and inherits a relative position relationship of text elements in the two records or more, thereby decreasing the number of hierarchical elements of the structured document; and

converting the structured document by replacing the two records or more with the new record.

11. A converting apparatus, comprising:

document inputting means for inputting information of a structured document that is written with a set of hierarchical elements, and composed of a plurality of records each including one text element or more;

joining means for generating a new text element by combining contents of text elements relatively at a same position among two records or more of the structured document to preserve a text relationship between the contents;

generating means for generating a new record that includes the text new element and inherits a relative position relationship of text elements in the two records or more;

converting means for converting the structured document by replacing the two records or more with the new record, thereby decreasing the number of hierarchical elements of the structured document; and

document outputting means for outputting the structured document after being converted.

12. A method of compressing a hierarchically structured document, comprising:

analyzing a hierarchy of the hierarchically structured document; and

combining hierarchical text elements of the hierarchically structured document responsive to common element names for the hierarchical text elements and a common child element structure of the hierarchical text elements and preserving a text relationship between contents of the text elements.

13. A method of compressing a hierarchically structured document, comprising:

analyzing a hierarchy of the hierarchically structured document; and

combining hierarchical text elements of the hierarchically structured document responsive to common features of parts of the hierarchy of the hierarchically structured document and preserving a text relationship between contents of the text elements.

14. A method of compressing a hierarchically structured document, comprising:

analyzing a hierarchy of the hierarchically structured document; and

compressing the hierarchy of the structured document by combining text element content and preserving a hierarchically defined relationship between contents of text elements.

IX. EVIDENCE APPENDIX

Not applicable.

X. RELATED PROCEEDINGS APPENDIX

Not applicable.